

COMPRESS Function Output

This chapter covers the following topics:

- Compressed Data Records
 - Rejected Data Records
 - ADACMP Report
-

Compressed Data Records

The data records that ADACMP has processed, edited, and compressed are written out together with the file definition information to a sequential dataset with the "variable blocked" record format. This dataset may be used as input to the ADALOD utility. The output of several ADACMP executions may also be used as input to ADALOD.

If the output dataset contains no records (no records provided on the input dataset or all records rejected), the output may still be used as input to the ADALOD utility. In this case, you must ensure that the amount of Associator space allocated to the file is sufficient since an accurate estimate cannot be made by the ADALOD utility without a representative sample of input record values (see the ADALOD utility for additional information).

Rejected Data Records

Any records rejected during ADACMP editing are written to the DD/FEHL error dataset. The records are output in variable blocked format and have the following structure:

Bytes	Description
0-1	Record length in binary format
2-3	Set to zero (X'0000')
4-5	Field name as stored in FDT
6-7	Offset from beginning of input record to error value
8-11	Input record sequence number (the first input record is "1")
12	PE index (if applicable)
13	Adabas response code (in hexadecimal)*
14-15	(reserved; set to zeros)
16	DD/EBAND input record

* Additionally the following response codes may occur:

X'E7'(231)	Input record too short (COMPRESS)
X'E8'(232)	Output record length error (COMPRESS)

Only the first incorrect field within a record is detected and referenced. If there are other errors, they are not detected until subsequent runs are made.

Example of Rejected Data Records

Field Definitions:

ADACMP FNDEF='01,AA,3,A,DE'

ADACMP FNDEF='01,AB,2,U'

ADACMP FNDEF='01,AC,3,P,NU'

Input record values (shown in hexadecimal): ISN = 3849 (decimal)

C1C2C3404000000F

1-3 4-5 6-8

Rejected record as output by ADACMP (shown in hexadecimal):

0018 0000 C1C2 0003 00000F09 00 37 0000 C1C2C3404000000F

0-1 2-3 4-5 6-7 8-11 12 13 14-15 16-23

The error dataset may be printed using the standard print utility provided with the operating system in use at the user installation. OS/390 or z/OS users may use the IEBPTPCH utility. VSE/ESA users may use the DITTO program. RDW (record descriptor word, bytes 1-4) may or may not be present, depending on the print utility used.

ADACMP Report

ADACMP calculates the approximate amount of space (in both blocks and cylinders) required for Data Storage for the compressed records. This information is printed as a matrix which contains the required space for the different device types requested by the DEVICE parameter for various Data Storage padding factors between 5 and 30 percent.

The following is an example of ADACMP report output:

PARAMETERS:

ADACMP COMPRESS NUMREC=1000
 ADACMP FNDEF='01,AA,8,B,DE'
 ADACMP FNDEF='01,BA,6,A,NU'
 ADACMP FNDEF='01,BB,8,P,NU'
 ADACMP FNDEF='01,AD,1,A,FI'
 ADACMP SUBDE='CA=BA(1,3)'

COMPRESS PROCESSING STATISTICS:

NUMBER OF RECORDS READ	1,000
NUMBER OF INCORRECT RECORDS	0
NUMBER OF COMPRESSED RECORDS	1,000
RAW DATA	24,000 BYTES
COMPRESSED DATA	16,656 BYTES

```

COMPRESSION RATE                31.9 %
LARGEST COMPRESSED RECORD        20 BYTES

```

DATA STORAGE SPACE REQUIREMENTS:

DEVICE	PADDING	BLOCKSIZE	NUMBER OF	
FACTOR	BYTES	BLOCKS	CYLS	
3380		4,820		
5%	4,578	4	1	
10%	4,337	4	1	
15%	4,096	5	1	
20%	3,856	5	1	
25%	3,615	5	1	
30%	3,373	5	1	

TEMP SPACE ESTIMATION:

I	DEVICE	I	BLOCKSIZE	I	NUMBER OF	I
I		I	BYTES	I	BLOCKS	CYLS
I		I		I		
I	3380	I	7,476	I	5	1
I		I		I		

THE LARGEST DESCRIPTOR IS AA, IT WILL OCCUPY 1 TEMP BLOCKS

SORT SPACE ESTIMATION:

I	DEVICE	I	BLOCKSIZE	I	LWP	I	NR OF	I
I		I	(BYTES)	I	(BYTES)	I	BLOCKS	CYLS
I	3380	I	7476	I	139264 (MINIMUM)	I	2	1
I		I		I	1048576 (DEFAULT)	I	2	1
I		I		I	139264 (OPTIMUM)	I	2	1
I		I		I		I		

The compression rate is computed based on the real amount of data used as input to the compression routine. Fields skipped by a format element "nX" (used to fill a field with blanks) are not counted.